AMENDMENTS TO THE CLAIMS

The listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Previously Presented) A computer-implemented method to load objects in a heterogeneous multiprocessor computer system, said method comprising:

   analyzing a source program for one or more program characteristics, the program characteristics selected from the group consisting of data locality, computational intensity, and data parallelism;

   in response to the analyzing, compiling the source program into two object files, a first object file corresponding to a first instruction set and a second object file corresponding to a second instruction set;

   in response to the compiling, storing the program characteristics in each of the object files;

   receiving a request to execute a software task corresponding to the source program;

   selecting a processor from a plurality of dislike processors, wherein a first processor corresponds to the first instruction set and a second processor corresponds to the second instruction set, to execute the software task, the selecting comprising comparing one or more characteristics of the software task with the program characteristics stored in the first object file and the second object file;

   in response to selecting the first processor:

      loading the first object file into a shared memory, wherein the shared memory is shared by the plurality of dislike; and executing the loaded first object file by the first processor; and

in response to selecting the second processor:

loading the second object file into the shared memory; and

executing the loaded second object file by the second processor.

2. (Canceled)

3. (Canceled)

4. (Canceled)

5. (Previously Presented) The method as described in claim 1 wherein selecting the processor further comprises:

retrieving the program characteristics;

retrieving current system characteristics, wherein the current system characteristics includes processor load characteristics for the plurality of dislike processors; and

combining the program characteristics and the current system characteristics to determine which of the dislike processors to assign the software task.

6. (Original) The method as described in claim 5 wherein at least one of the current system characteristics is selected from the group consisting of processor availability for each of the dislike processors, and a data size of data being processed by the software task.

7. (Previously Presented) The method as described in claim 1 wherein executing the first object file by the first processor further comprises:

determining that the first processor has a scheduler that schedules tasks for the first processor; and

scheduling the first object file to execute on the first processor, the scheduling including:

> writing a software code identifier corresponding to the first object file to a run queue corresponding to the first processor.

8. (Previously Presented) The method as described in claim 1 wherein executing the second object file by the second processor further comprises:

signaling the second processor; and

reading, by the second processor, the second object file from the shared memory into a local memory corresponding to the second processor.

9. (Previously Presented) The method as described in claim 8 further comprising:

writing an instruction block in the shared memory, the instruction block including an address of the loaded second object file and an address of an input buffer; and

reading the loaded second object file and the input buffer from the locations identified in the instruction block to the second processor 's local memory.

10. (Previously Presented) The method as described in claim 9 further comprising:

signaling the second processor from one of the other processors, the signaling including:

> writing the address of the instruction block to a mailbox that corresponds to the second processor; and

reading, by the second processor, the instruction block in response to the signal.

11. (Canceled)

12. (Canceled)

13. (Canceled)

14. (Canceled)

15. (Canceled)

16. (Canceled)

17. (Canceled)

18. (Canceled)

19. (Canceled)

20. (Canceled)

21. (Canceled)

22. (Canceled)

23. (Canceled)

24. (Canceled)

25. (Canceled)

26. (Canceled)

27. (Canceled)

28. (Canceled)

29. (Canceled)

30. (Canceled)